UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/879,658	06/11/2001	Takahide Ohkami	264/087	2956

34313 7590 11/18/2004

ORRICK, HERRINGTON & SUTCLIFFE, LLP
4 PARK PLAZA
SUITE 1600
IRVINE, CA 92614-2558

EXAMINER	
SHARON, AYAL I	

ART UNIT	PAPER NUMBER
2123	

DATE MAILED: 11/18/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/879,658	OHKAMI, TAKAHIDE	
	Examiner	Art Unit	
	Ayal I Sharon	2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 June 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-9* is/are rejected.

7)☒ Claim(s) *10* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11 June 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *5/8/2003*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Introduction

1. Claims 1-10 of U.S. Application 09/879,658 filed on 06/11/2001 are presented for examination. The application claims priority to provisional application 60/242,407, filed on 10/20/2000.

Claim Objections

2. Claim 10 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and all intervening claims.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. The prior art used for these rejections is as follows:

5. Koch, G. et al. "Breakpoints and Breakpoint Detection in Source-Level Emulation." ACM Transactions on Design Automation of Electronic Systems. April 1998. vol.3, no.2, pp.209-230. (Henceforth referred to as "**Koch**").

6. Patel, Chandresh. U.S. Patent 5,546,562. (Henceforth referred to as "**Patel**").

7. The claim rejections are hereby summarized for Applicant's convenience. The detailed rejections follow.

8. **Claims 1-5 are rejected under 35 U.S.C. 102(b) as being anticipated by Koch.**

9. In regards to Claim 1, Koch teaches the following limitations:

1. A method for compiling a user's design to allow greater access for reading and writing to memories and registers in the user's design, comprising;

identifying all of the memories and registers in the user's design;
(See Koch, especially: p.210, para.3)

Koch teaches (emphasis added): "In Koch et al. [1995] we proposed a method called Source Level Emulation (SLE) to close this gap by combining the functionality of behavioral simulation with the speed of hardware emulation. The idea is to run the application on emulator hardware and keep track of the correlation between hardware elements and the behavioral VHDL source code, allowing the emulator's operation to be controlled at the source-code level and for register values read from the emulator to be related back to the source code.

Examiner finds that "keep[ing] track of the correlation between hardware elements and the behavioral VHDL source code" corresponds to "identify[ing] all the memories and registers in the user's design."

Examiner also finds that a "register" is a form of "memory".

synthesizing accessibility logic into the user's design,
(See Koch, especially: p.210, para.4)

Koch teaches (emphasis added): "SLE allows running hardware to be debugged symbolically, as in software debugging, including examining variables, setting breakpoints, and performing single step operations. All this is possible with the application running as a real hardware implementation on a hardware emulator. By back annotating the values read from the circuit, debugging can be done at the source-code level."

said accessibility logic creating access ports to the memories and registers.
(See Koch, especially: p.210, para.4)

Koch teaches (emphasis added): "and for register values read from the emulator to be related back to the source code. ... All this is possible with the application running as a real hardware implementation on a hardware emulator. By back annotating the values read from the circuit, debugging can be done at the source-code level."

Examiner finds that that the ability to read register values from an emulator by using a "real hardware implementation on that hardware emulator" inherently requires the use of access ports.

10. In regards to Claim 2, Koch teaches the following limitations:

2. The method of claim 1 further comprising the step of assigning a unique identifier to each of the memories and registers in the user's design..
(See Koch, especially: p.213, para.2)

Koch teaches: "Since SLE reads only the register contents from the emulated circuit, the value of a particular variable might have to be computed if the corresponding net for this variable is not assigned to a register."

Examiner finds that the "assignment" of variables to registers corresponds to linking variables to specific registers. This inherently requires unique identifiers for each of the registers / memories. (Examiner also finds that a "register" is a form of "memory".)

11. In regards to Claim 3, Koch teaches the following limitations:

3. The method of claim 2 wherein said accessibility logic comprises selecting logic, said selecting logic adapted to receive said unique identifier and select a particular one of the memories and registers in the user's design.
(See Koch, especially: p.213, para.2 and Fig.2)

Koch teaches: "If a net that corresponds to a variable is not assigned to a register, the value of the net can be determined according to the cases shown in Figure 2. ... In case 1, the net representing the requested variable (shown in boldface) is an input net of the component and is connected to a register via several multiplexers."

All four cases in Fig.2 show the use of multiplexers ("Mux"), which is a "selecting logic." Examiner finds that the use of unique ID's to select items (in this case, registers) is an inherent feature of multiplexors.

12. In regards to Claim 4, Koch teaches the following limitations:

4. The method of claim 3 wherein said accessibility logic comprises logic to read

from or write to said particular one of the memories and registers in the user's design.
(See Koch, especially: p.213, para.2 and Fig.2)

Koch teaches (emphasis added): "In case 1, the net representing the requested variable (shown in boldface) is an input net of the component and is connected to a register via several multiplexers. Only the register has to be read to retrieve the value of this net."

13. In regards to Claim 5, Koch teaches the following limitations:

5. The method of claim 4 wherein said accessibility logic comprises decode logic that receives commands from a host and controls execution of reading and writing data to the memories and registers in the user's design.
(See Koch, especially: p.213, para.2 and Fig.2)

All four cases in Fig.2 show the use of multiplexers ("Mux"), which is an "accessibility logic" that "controls reading and writing data to the memories and registers in the user's design". Examiner finds that the use of unique ID's to select items (in this case, registers) is an inherent feature of multiplexors.

Claim Rejections - 35 USC § 103

14. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

15. The prior art used for these rejections is as follows:

16. Koch, G. et al. "Breakpoints and Breakpoint Detection in Source-Level Emulation." ACM Transactions on Design Automation of Electronic Systems. April 1998. vol.3, no.2, pp.209-230. (Henceforth referred to as "**Koch**").

17. Patel, Chandresh. U.S. Patent 5,546,562. (Henceforth referred to as "**Patel**").

18. "X.25". Internet Technology Overview, June 1999. Chapter 17. (Henceforth referred to as "**X.25**").

19. The claim rejections are hereby summarized for Applicant's convenience. The detailed rejections follow.

20. Claims 6-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Koch in view of Patel.

21. In regards to Claim 6, Koch teaches the following limitations:

6. A hardware-assisted design verification system for verifying a target design, said verification system having a host workstation in communication with a hardware accelerator, the target design comprising registers and memories, the host workstation loading data to or unloading data from the registers and memories, comprising:

command decode logic, said command decode logic decoding a command in said incoming packet register to identify a particular operation, register or memory location in said target design;
(See Koch, especially: pp.223-225 "5.2. Debugging Controller", and Fig.9, "run circuit" item in bottom left corner)

write command execution logic to write data stored in said incoming packet register into said register or memory location in said target design for a write command decoded at said command decode logic;
(See Koch, especially: pp.223-225 "5.2. Debugging Controller", and Fig.9, "write to breakpoint" item in top left corner)

read command execution logic to read data from said register or memory location in said target design and store said data in said outgoing packet register for a read command decoded at said command decode logic; and
(See Koch, especially: pp.223-225 "5.2. Debugging Controller", and Fig.9, "read data path registers" item in bottom right corner)

interface logic interfacing said registers and memories in said target design.
(See Koch, especially: pp.223-225 "5.2. Debugging Controller", and Figs.9 and 10a)

Koch teaches (p.224): "Each register in the data path is replaced as shown in Figure 10 by a register that allows the required control of the operation."

However, Koch does not expressly teach the following limitations:

protocol logic synthesized into the logic circuit, said protocol logic comprising:
an incoming packet register in communication with said host workstation;
an outgoing packet register in communication with said host workstation;

Patel, on the other hand, does expressly teach the use of "Hardware Buffers and Latches" that store incoming and outgoing data signals in the communication between a "Reference Element" (which corresponds to the claimed "protocol logic") and an "Arbitration and Synchronization Cicuit" (which corresponds to the claimed "host workstation"). See Items 120 and 124 in Figure 9A and col.23, line 63 to col.24, line 14.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the teachings of Koch with those of Patel, because the Patel reference "... provide[s] a way to integrate the use of emulation systems ... into logic simulation and analysis, such that a design team can take a net list for a VLSI device, load it onto an emulation systems and use the proposed emulation modeling apparatus to provide visibility and control within the ASIC being designed." (Patel: col.5, lines 10-16).

22. In regards to Claim 7, Patel teaches the following limitations:

> 7. The hardware-assisted design verification system of claim 6, wherein said protocol logic includes logic to determine whether data from said incoming packet register is new and control activation of command decoding and execution.

(Patel: Fig.9B, Items "Breakpoint Hit?" and "Update GUI?" and col.24, lines 4-14).

23. Claim 8 is rejected under 35 U.S.C. 103(a) as being unpatentable over Koch in view of X.25.

24. In regards to Claim 8, Patel teaches the following limitations:

> 8. A method of synthesizing a packet-based protocol logic for providing access to registers and memories in a target design when performing functional verification using a hardware accelerator, comprising:
>
> counting how many of the registers are present in the target design;

(See Koch, especially: p.210, para.3)

Koch teaches (emphasis added): "In Koch et al. [1995] we proposed a method called Source Level Emulation (SLE) to close this gap by combining the functionality of behavioral simulation with the speed of hardware emulation. The idea is to run the application on emulator hardware and keep track of the correlation between hardware elements and the behavioral VHDL source code, allowing the emulator's operation to be controlled at the source-code level and for register values read from the emulator to be related back to the source code.

Examiner finds that "keep[ing] track of the correlation between hardware elements and the behavioral VHDL source code" corresponds to "identify[ing] all the memories and registers in the user's design."

counting how many of the memories are present in the target design;
(See Koch, especially: p.210, para.3)

Examiner finds that a "register" is a form of "memory".

determining a maximum number of data bits of the registers in the target design;
(See Koch, especially: p.210, para.3)

Examiner finds this to be inherent to when reading values from a register – otherwise it is not possible to correctly read the values.

determining a maximum number of data bits of the memories in the target design;
(See Koch, especially: p.210, para.3)

Examiner finds that a "register" is a form of "memory", and therefore this limitation to be inherent to when reading values from a register – otherwise it is not possible to correctly read the values.

determining a maximum number of address bits of the memories in the target design; and
(See Koch, especially: p.210, para.3)

Examiner finds that a "register" is a form of "memory", and therefore this limitation to be inherent to when reading values from a register – otherwise it is not possible to correctly read the values.

determining a maximum number of bits to send the register data, memory data, and memory address to the target design to determine data field size of said request packet.
(See Koch, especially: p.210, para.3)

Examiner finds that a "register" is a form of "memory", and therefore this limitation to be inherent to when reading values from a register – otherwise it is not possible to correctly read the values.

However, Patel does not expressly teach the following limitations. X.25 on the other hand, does expressly teach these limitations:

determining fixed sizes of a request packet, said request packet comprising tag, command, and data end fields;
(X.25, especially: p.17-2, "Packet Assembler / Disassembler" and Fig.17-2; also p.17-6 "LAPB Frame Format" and Fig.17-6)

determining a maximum identification field size of said request packet;
(X.25, especially: p.17-6 "LAPB Frame Format" – "Control" field and Fig.17-6)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the teachings of Koch with those of X.25, because the Koch reference teaches more effective testing techniques for products like those taught in the X.25 reference. The Koch reference teaches a method that "... allows source-level debugging of running hardware specified in behavioral VHDL in a way similar to source-level debugging in software programming languages," (Koch, Abstract), while the X.25 reference teaches the use of a Packet Assembler / Disassembler hardware product which "... performs three primary functions: buffering, packet assembly, and packet disassembly." (X.25, p.17-2).

25. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Koch in view of X.25 and further in view of Patel.

26. In regards to Claim 9, Koch teaches the following limitation:

creating interface logic to access the registers and memories in said target design.
(See Koch, especially: pp.223-225 "5.2. Debugging Controller", and Figs.9 and 10a)

> Koch teaches (p.224): "Each register in the data path is replaced as shown in Figure 10 by a register that allows the required control of the operation."

Koch does not expressly teach the following limitations. Patel, on the other hand, does teach them:

9. The method of claim 8, further comprising the steps of:

creating an incoming packet register coupled to an input data buffer in the hardware accelerator;
(Patel, especially: Fig.9a, Item 120. "Hardware Buffer"; col.23, line 55 to col.24, line 59)

creating an outgoing packet register coupled to an output data buffer in the hardware accelerator;
(Patel, especially: Fig.9a, Item 124. "Hardware Buffer"; col.23, line 55 to col.24, line 59)

creating a command decode block to decode a command in said incoming packet register;
(Patel, especially: Fig. 9a, Item 118. "Arbitration & Synchronization Control Circuit"; col.23, line 55 to col.24, line 59)

creating an execution logic to execute a command decoded at said decode block; and
(Patel, especially: Fig. 9a, Item 122. "Reference Element"; col.23, line 55 to col.24, line 59)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the teachings of Koch with those of Patel, because the Patel reference "... provide[s] a way to integrate the use of emulation systems ... into logic simulation and analysis, such that a design team can take a net list for a VLSI device, load it onto an emulation systems and use the proposed emulation modeling apparatus to provide visibility and control within the ASIC being designed." (Patel: col.5, lines 10-16).

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ayal I. Sharon whose telephone number is (571) 272-3714. The examiner can normally be reached on Monday through Thursday, and the first Friday of a biweek, 8:30 am – 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kevin Teska can be reached at (571) 272-3716.

Any response to this office action should be faxed to (703) 872-9306 or mailed to:

Director of Patents and Trademarks
Washington, DC 20231

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Tech Center 2100 Receptionist, whose telephone number is (571) 272-2100.

Ayal I. Sharon

Art Unit 2123

November 10, 2004